Computer-aided ladder crystal filter design

by J. A. HARDCASTLE, G3JIR*

Introduction

Ever since the author's first experiments with ladder crystal filters, the principal objective has been to produce simple procedures for use by radio amateurs wishing to make filters of predictable performance. This was largely achieved, with a minimum of complicated mathematics, by using the Amstutz component coefficients [1] which allowed the design to be completed using a slide-rule or pocket calculator. During the intervening period personal computers have become generally available, so mathematical complexity need no longer be a constraint once the required programs have been written.

The procedure to be described has been developed from Dishal's design [2] to allow filters to be constructed from available inexpensive surplus crystals. This is the reverse of the usual procedure where the professional filter engineer calculates the parameters for the required crystals before ordering them to be made to his design.

Crystal parameters

Before the performance of a crystal filter can be calculated the electrical properties of the crystals themselves must be known. Resonance of a crystal depends on the mechanical properties of quartz but it can be represented by the equivalent electrical circuit shown in Fig 1. This shows that it possesses both a series resonance and, at a slightly higher frequency, a parallel resonance.

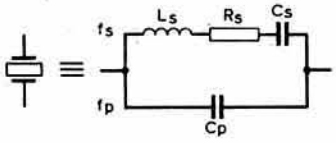

Fig 1. Equivalent circuit of a crystal

The shunt capacitance C_p may be measured on a component bridge, and is principally the capacitance of the crystal's electrodes and case assembly.

The equivalent series capacitance C_s cannot be measured directly, and must be derived from measurements made in the test circuit of Fig 2 which is loosely based on British Standard BS9610. Tuning the signal generator through the series resonant frequency f_s gives a peak on the signal detector, and a little higher in frequency a sharp null is found at the parallel resonance f_p giving a frequency response similar in shape to Fig 3. Initially S1 is set to position 1, and the series resonance f_{s1} measured. Similarly f_{s2} and f_{s3} are measured with S1 set to positions 2 and 3. Series resonance in these measurements is a rather broad peak, and if a double-beam, wide-band, oscilloscope is available this may be connected as shown in Fig 2 and a more precise indication of resonance obtained by tuning to the point where both signals are in phase.

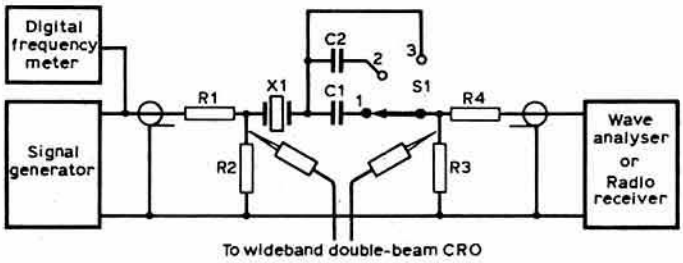

Fig 2. Crystal test circuit

*8 Norwood Grove, Rainford, St Helens, Merseyside WA11 8AT.

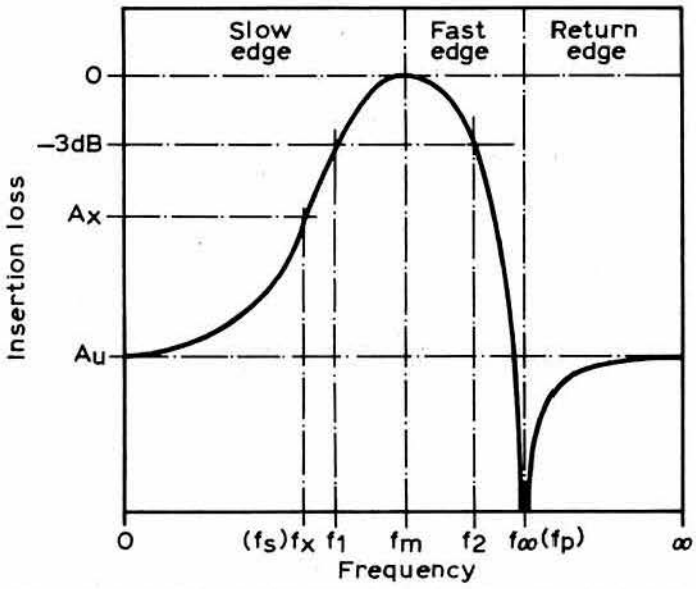

Fig 3. Generalized shape of ssb crystal filter showing the principal regions and frequencies referred to in the text

The equivalent series capacitance may now be calculated from

$$C_S = \frac{2(C2-C1)\ 10^{-12}}{f_{s3}} \times \frac{(f_{s1}-f_{s3})(f_{s2}-f_{s3})}{f_{s1}-f_{s2}} \qquad (1)$$

where C = pF and f = Hz

Typical values measured on nominally 9,681·2kHz crystals were:

f_{s1} = 9,684,598Hz; f_{s2} = 9,681,128Hz; f_{s3} = 9,676,669Hz

C1 = 9·61pF; C2 = 25·32pF,

which gives $C_S = 3{\cdot}308 \times 10^{-14}$ or 0·033pF.

For this measurement R1 and R4 were 1,000Ω, and R2 and R3 were 220Ω. The average value of series resonance for the batch of crystals was 9,677,200Hz.

The choice of R2 and R3 is a compromise since series and parallel resonance are both affected by stray capacitance. With R2 and R3 both 15Ω the parallel resonant frequency was 9,681kHz. This increased steadily as the impedance was increased, and eventually levelled off to 9,694kHz at 220Ω; the series resonance hardly changing up to this point. Ideally the measurements should be made at the same impedance as the final filter but this obviously cannot be known at this stage in the design.

Careful screening between input and output of the test circuit is essential because unwanted signal feedthrough and noise broaden the signal null at f_p, reducing the precision of the measurement.

The shunt capacitance was measured as 8·3pF and can be used in (2) to crosscheck the measurement of f_p.

$$f_p = f_s\sqrt{1+(C_s/C_p)} \qquad (2)$$
$$= 9{,}677{,}200\sqrt{1+(0{\cdot}03/8{\cdot}3)}$$
$$= 9{,}694{,}673\text{Hz}$$

Single sideband frequency response

Fig 3 shows a typical lower sideband filter response. For ease of description and calculation it has been divided into three distinct regions which Dishal has named "slow", "fast" and "return" edges. Fig 3 also shows the principal frequencies of interest, and it should be noted that for a lower sideband filter f_s coincides with f_x, and f_p with f_∞.

Upper sideband filters will not be considered here since to do so would double the length of the article, but it should be noted that for the usb case the frequency response is a mirror image of Fig 3, and may be calculated

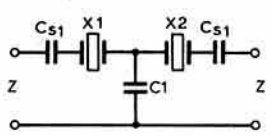

Fig 4. Two-pole filter. Table 1 gives components and other details

Table 1. Two-pole Butterworth filters

Fig. 5	Z (Ω)	Cl (pF)	fm (Hz)	BW (Hz)
a	658	25	9,683,163	6,266
b	328	50	9,681,084	4,685
c	164	100	9,679,518	3,017
d	55	300	9,678,091	1,225

Table 2. Attenuation at crystal's series resonant frequency

Butterworth filters		1dB Chebyshev filters	
Filter order	Attenuation A_x (dB)	Filter order	Attenuation A_x (dB)
2	7·0	2	9·3
3	9·5	3	11·4
4	11·6	4	12·3
6	29·9	6	19·9
8	56·5	8	29·0

Table 3. Dishal coupling coefficients

Butterworth filters

n	k_{12}	k_{23}	k_{34}	k_{45}	k_{56}	k_{67}	k_{78}	d	K
2	0·7071							0·7071	1·4142
3	0·7071	0·7071						1·0	1·4142
4	0·841	0·541	0·841					1·307	1·382
6	1·169	0·605	0·518	0·605	1·169			1·932	1·774
8	1·519	0·736	0·554	0·510	0·554	0·736	1·519	2·563	2·255

1dB Chebyshev filters

n	k_{12}	k_{23}	k_{34}	k_{45}	k_{56}	k_{67}	k_{78}	d	K
2	0·735							0·451	1·47
3	0·644	0·644						0·451	1·288
4	0·638	0·546	0·638					0·452	1·184
6	0·633	0·531	0·520	0·531	0·633			0·453	1·164
8	0·633	0·530	0·514	0·511	0·514	0·530	0·633	0·454	1·163

by reversing the sign preceding the brackets in equations (9) to (11) which are given later. It must also be noted that for the usb filter f_p provides f_x, and f_x produces the notch frequency f_∞.

The simplest ladder crystal filter is shown in Fig 4, and its component values are listed in Table 1. It uses only two crystals and, as Fig 5 shows, it has very limited selectivity. However, this is included because it exhibits all the properties of the more complex higher-order filters at levels which are readily measured. Fig 5 also shows how the filter behaves when the impedance is altered. As the impedance is reduced, not only does the bandwidth decrease but the whole passband is shifted lower in frequency. This is a complication which anyone planning a switched bandwidth filter must take into account.

A striking feature of Fig 5 is the manner in which the four frequency responses coincide at the crystal's series resonant frequency, which acts as a pivot. This applies to all orders and types of filter and is a useful datum for sketching approximate frequency responses. Table 2 gives data for other filters.

Bandwidth and centre frequency

As can be seen from Fig 5, bandwidth B depends on the position of the midband frequency f_m in relation to f_x and f_∞, and designers need a ready means of determining f_m for a given bandwidth. This is provided by the following formula:

$$(B/2)^2 - BK(f_\infty - f_x)/2 - (f_x - f_m)(f_\infty - f_m) = 0 \quad (3)$$

Solving for f_m gives

$$f_m = f_x + 0{\cdot}5(T - \sqrt{T^2 - 2BKT + B^2}) \quad (4)$$

where $T = f_\infty - f_x$

and $K = k_{12} + k_{23}$ (see Table 3)

Maximum bandwidth is obtained when f_m is midway between f_x and f_∞ and may be calculated from

$$B_{max} = (f_\infty - f_x)[K - \sqrt{K^2 - 1}] \quad (5)$$

The above relationships show that bandwidth reduces in a parabolic manner as f_m approaches either f_x or f_∞, but in practice only the half of the characteristic on the side of f_x produces realistic designs.

Computer program 1 (FM6) solves the above equations to provide an output of maximum possible bandwidth, and bandwidth for a given f_m, when provided with input data consisting of f_x, f_∞ and K.

As a further aid Fig 6 is a universal design chart which relates normalized bandwidth W to normalized frequency P. To use the chart, W is calculated from B as follows,

$$W = B/(f_\infty - f_x) \quad (6)$$

P is then read off the appropriate curve and f_m is calculated from

$$f_m = f_x + P(f_\infty - f_x) \quad (7)$$

The following example illustrates this procedure.

Measured f_∞ = 9,694,000Hz and f_x = 9,677,200Hz

For a required bandwidth B = 3,017Hz. W = 0·1796 is obtained from (6). For a two-pole Butterworth filter the chart gives P = 0·14 and from (7) f_m = 9,679,552Hz.

By comparison, program FM6 gives the value of f_m as 9,679,518Hz and its measured value was 9,679,544Hz at a measured bandwidth of 2,977Hz.

Frequency response

Having obtained the midband frequency, the program SSBBPF together with FRBUTT or FRCHEBY are used to calculate the filter frequency response. They apply the following formulas.

Firstly the frequency-bandwidth ratio R is calculated from

$$R = 2(f_\infty - f_m)/B \quad (8)$$

Then the ssb response is calculated in the three separate regions of Fig 3 using Dishal's frequency transformation [2] given in (9), (10) and (11).

Slow edge $$f(A) = f_m - \left\{\frac{B}{2} \times \frac{XR - 1}{R - X}\right\} \quad (9)$$

Return edge $$f(B) = f_m + \left\{\frac{B}{2} \times \frac{XR - 1}{X - R}\right\} \quad (10)$$

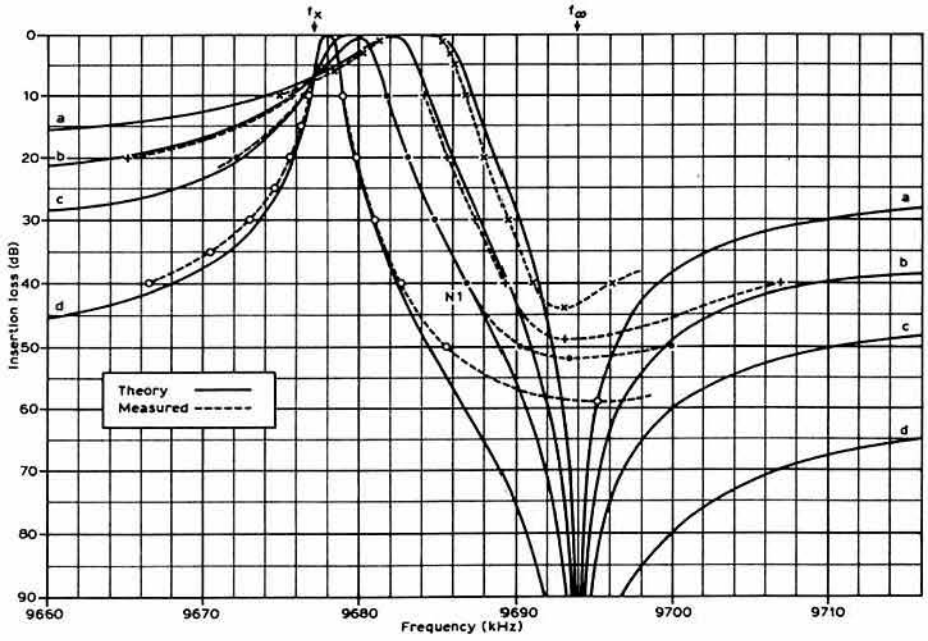

Fig 5. Two-pole Butterworth filter frequency response for the circuit of Fig 4

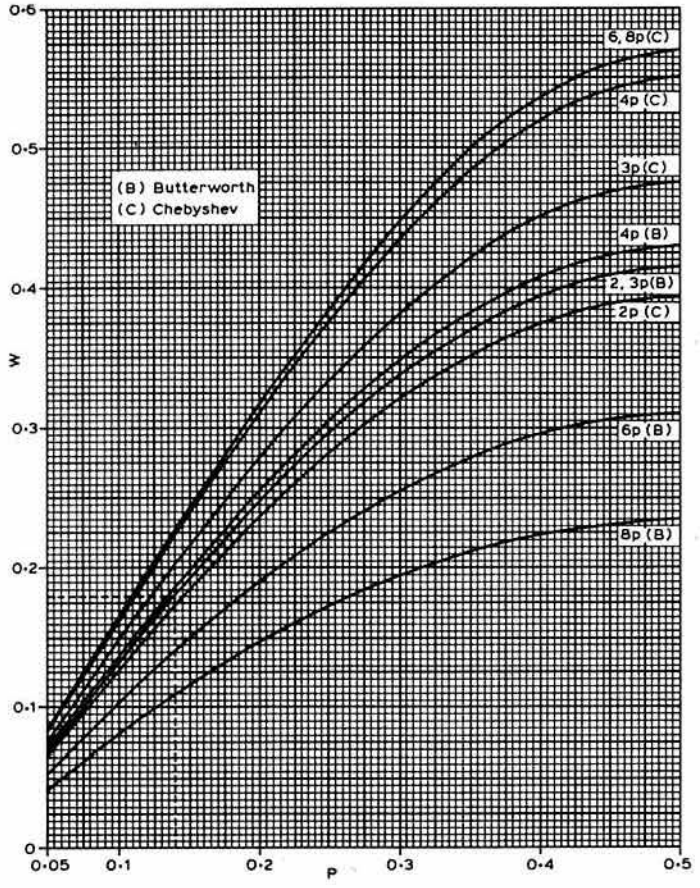

Fig 6. Universal design chart

Fast edge $$f(C) = f_m + \left\{\frac{B}{2} \times \frac{XR+1}{X+R}\right\} \quad (11)$$

where X is the normalized frequency ratio

$$X = f/f_c \quad (12)$$

and where f_c is the lpf −3dB cut-off frequency and f is the actual frequency.

This may be read off either Fig 7 or Fig 8 according to filter type.

Using B = 3,017Hz and f_m = 9,679,518Hz (8) gives a value for R = 9·60.

From Fig 7 it is found that at X = 10 the insertion loss is 40dB. Using (11) it is calculated that at −40dB f(C) = 9,686,984Hz. This is shown marked N1 on Fig 5.

As mentioned previously, the frequency f_x acts as a pivot for the frequency response and it is worth noting that at f_x

$$X = k_{12} + k_{23} = K \quad (13)$$

The ultimate attenuation reached at points far removed from the passband is attained when X is equal to R. Reading from Fig 7 gives an attenuation of 39dB.

To use Program 2, the main program SSBBPF is merged with FRBUTT or FRCHEBY according to filter type. For each value of attenuation entered, the program will calculate the corresponding frequency in the three regions of the filter. The "slow" and "return" edges will return zero values when the attenuation requested is out of their range. The frequency response calculated by FRBUTT is shown in Fig 5(c).

Transmitter filters

Using the value of f_p obtained in the initial measurements ensures the best approximation to a symmetrical frequency response for a receiver. However, a transmitter filter needs the steepest possible rate of cut-off and the best attainable discrimination to the unwanted sideband. This type of response is obtained by shunting additional capacitance C_p across the crystals to bring f_∞ closer to the passband. However, this results in a diminished rate of cut-off of the "slow edge", but this can be compensated by an audio filter to limit the bandwidth.

Having decided on a desirable value for f_∞, the previous procedure is repeated to find the new value of f_m. Equation (8) is again used to obtain R, and the crystal shunt capacitance is calculated from

$$C_p = C_s\left[\frac{f_m}{B}\right]\frac{R-K}{R^2-1} \quad (14)$$

The shunt capacitance obtained includes the crystal holder and stray capacitance measured previously, and this must be subtracted from C_p to determine the amount to be added. In practice the shunt capacitor may be made a trimmer which is adjusted to bring the attenuation notch to the required frequency.

This calculation is included in Program 3 (DISHAL) at line 1170.

Calculating component values

Having determined f_m and C_s, program DISHAL is run to obtain filter component values. It uses the following equations which are included for users of pocket calculators.

The reactance of the crystal at its series resonance is given by

$$X_x = \frac{1 \times 10^{12}}{2\pi f_s C_s} \quad (15)$$

where f = Hz and C = pF.

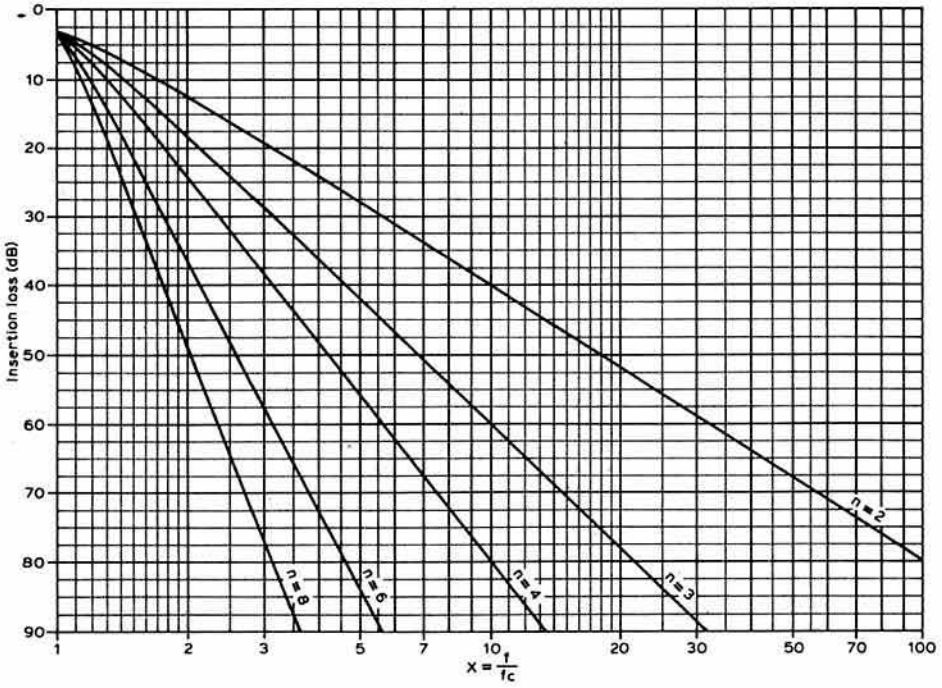

Fig 7. Butterworth lowpass filter. Attenuation/ normalized frequency response

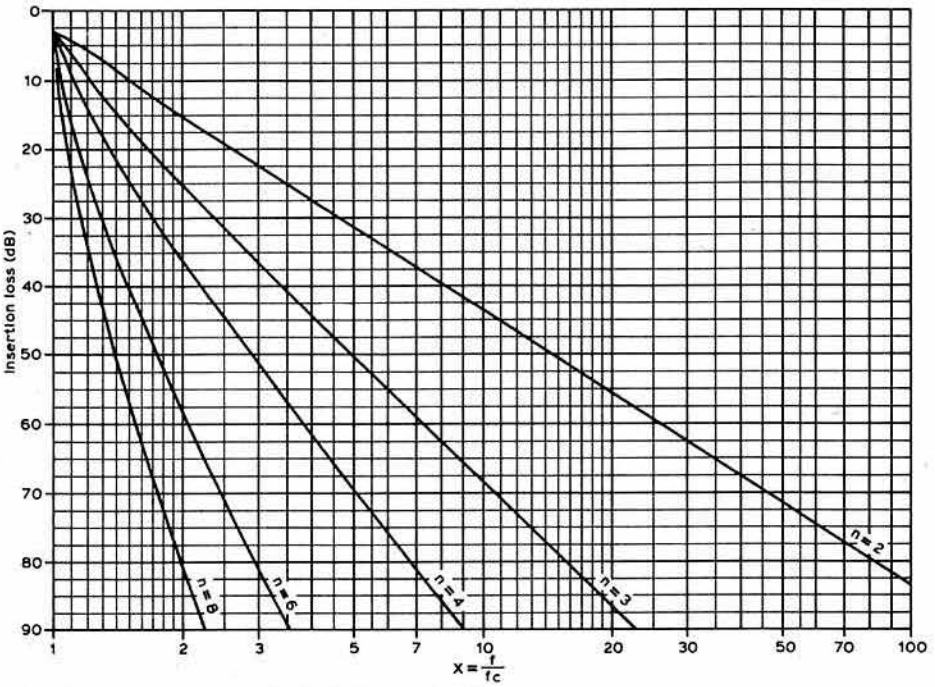

Fig 8. Chebyshev lowpass filter, 1dB passband ripple. Attenuation/normalized frequency response

This is modified next by a factor to allow for filter order and type

$$X_o = \frac{X_x}{[1-(K/R)]^2}. \quad (16)$$

The impedance is now calculated from

$$Z = d_1\left[\frac{X_o B\,(1-1/R^2)}{f_m}\right]\Omega. \quad (17)$$

Coupling capacitor values are then obtained from

$$X_1 = k_{12}\left[\frac{X_o B\,(1-1/R^2)}{f_m}\right] \quad (18)$$

$$C_1 = \frac{1\times 10^{12}}{2\pi f_m X_1}\text{pF}. \quad (19)$$

The other capacitor values are obtained by substituting the appropriate value of k_{nm} from Table 3 into (18).

Fig 9 gives the filter circuit and component designations.

d and k calculation

Throughout this article the Chebyshev designs have been calculated for 1dB passband ripple, but since other designers may want to use a different value the following formulas are included. They have also been used in Program 4 (DANDK) which will be found useful for calculating the variety of d's and k's used to design variable bandwidth filters of the type described by G3UUR [3]. These were derived by Dishal [4].

$$r = \text{passband ripple(dB)}/10 \quad (20)$$

$$E = 1/\sqrt{10^r - 1} \quad (21)$$

$$S = \sinh\left[\frac{1}{n}\sinh^{-1}E\right] \quad (22)$$

$$W = \cosh\left[\frac{1}{n}\cosh^{-1}E\right] \quad (23)$$

$$Q = 2\sin(\pi/2n)W/S. \quad (24)$$

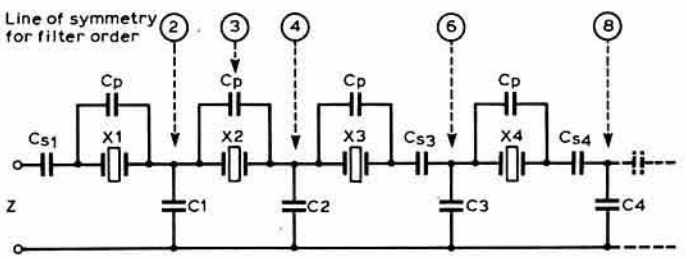

Fig 9. Generalized circuit diagram for ladder crystal filters. Filters are symmetrical about the line indicated for each order of filter

The dissipation coefficient is:

$$d = 1/Q. \quad (25)$$

The coupling coefficients are derived from:

$$k_{c,c+1} = \sqrt{\frac{S^2+\sin^2(c\pi/n)}{4\sin[(2c-1)\pi/2n]\sin[(2c+1)\pi/2n]}}\times\frac{1}{W} \quad (26)$$

where n = filter order

and c = 1, 2, – – –, n – 1.

Summary of design procedure

1. Measure f_s, f_p and C_p and calculate C_s from (1).
2. Use program FM6 to calculate f_m for the required bandwidth.
3. Use program SSBBPF with either FRBUTT or FRCHEBY to calculate the frequency response.
4. Use DISHAL to calculate filter component values.

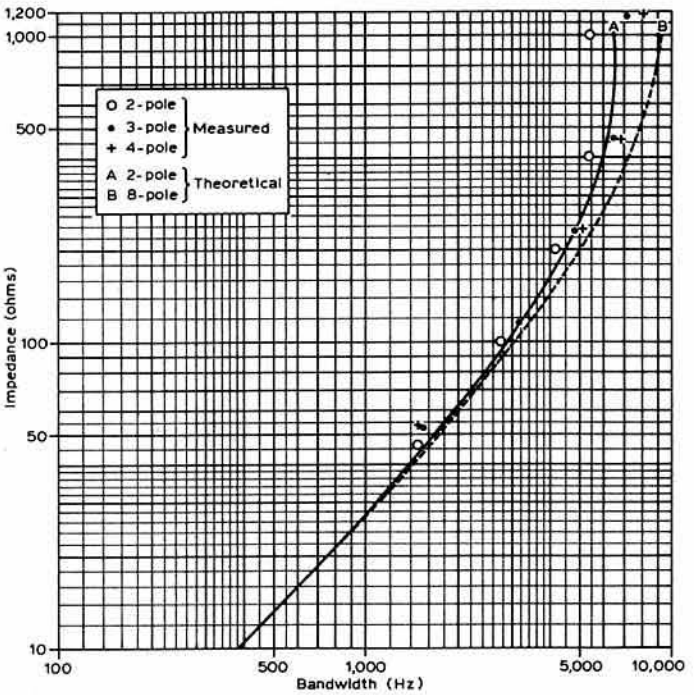

Fig 10. 1dB ripple Chebyshev filters. Impedance/bandwidth chart

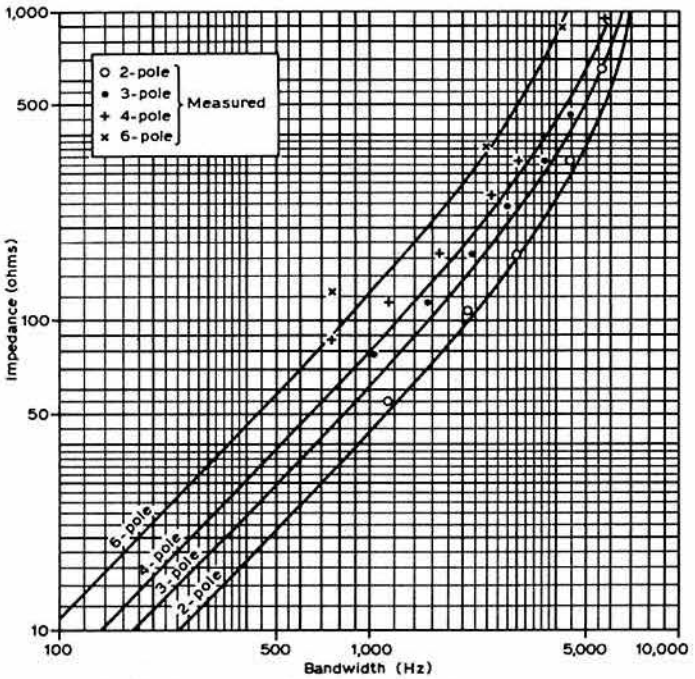

Fig 11. Butterworth filters. Impedance/bandwidth chart

Retrospect—Impedance and bandwidth

It has previously been found experimentally that for 1dB ripple Chebyshev filters, and for a given bandwidth, impedance is constant for all orders of filter. This important result was used as a means of rapidly extrapolating the design of high-order filters from measurements on second-order filters [1]. This has now been confirmed theoretically, as Fig 10 shows, and except at the very widest bandwidths there is very little difference between the bandwidths of second and eighth-order filters. Experimental results are also plotted on Fig 10, showing the good measure of agreement between theory and practice.

Unfortunately this simple relationship does not hold for Butterworth or low ripple Chebyshev filters, and the reason can be found in equation (17) and Table 3. Impedance can be seen to be proportional to the dissipation constant d and, while this remains almost the same for all orders of Chebyshev (1dB ripple) filters, it varies over a wide range for Butterworth filters as shown in Fig 11. This also shows the less satisfactory agreement between theory and practice for Butterworth designs. This has been investigated for second-order filters by using a general-purpose network simulation program, and this has shown that stray capacitance between ground and the junction of the crystal and series capacitor can account for these discrepancies. The lower impedance levels required by Chebyshev filters makes them less vulnerable to the effects of stray capacitance, and therefore they are usually more appropriate for amateur applications.

Conclusion

This article has described in detail a procedure for designing lower-sideband ladder crystal filters. Although the computer programs may be used without prior knowledge, sufficient supporting theory is provided to satisfy the requirements of pocket calculator users. Also for these readers there are graphical design aids and tabular data to assist them. It should be noted that while computers will provide greater digital accuracy than graphs, overall accuracy is limited by the precision of initial measurements of crystal parameters, and time spent on achieving this will be well rewarded.

Finally, attaining the theoretical performance depends on observing good hf construction practice, keeping high impedance points as far away from ground as possible, and screening each section from its neighbours.

Acknowledgements

The contribution of Mr M. Dishal as originator of most of the formulas used in this article cannot be too greatly emphasized and is gratefully acknowledged.

Also John Evans, G8IIQ, is thanked for examining the article at an early stage and for making constructive suggestions for improving its clarity.

It is less easy to thank the RSGB referees, as they must necessarily remain anonymous. However, they have succeeded in influencing this article beneficially, particularly by contributing simpler solutions for equations (4) and (5) used in program FM6. This replaces the author's more complex and longer-running successive approximation routine. Their great expenditure of time and effort is sincerely appreciated.

References

[1] "Ladder crystal filter design", J.A.Hardcastle, G3JIR. *Rad Com* February 1979, p116.

[2] "Modern network theory design of single-sideband crystal ladder filters", M.Dishal. *Proc IEEE* Vol 53 No 9 September 1965, pp1205-16.

[3] "Switched bandwidth ladder crystal filter", G3UUR. "Technical Topics", *Rad Com* December 1980, p1294.

[4] "Two new equations for the design of filters", M.Dishal, *Electrical Communication* December 1953, pp324-37.

Program 1. FM6 f_m calculator

```
]LIST

1  PRINT "FM6/9"
10  REM  CALC FM GIVEN BW
20  INPUT "FX     (HZ) = ";F2
30  INPUT "FINF   (HZ) = ";F9
40  INPUT "K (TABLE 3) = ";K
50 T1 = F9 - F2
60 B3 = T1 * (K -  SQR (K * K - 1))
70  PRINT : PRINT
80  PRINT "MAX BW      = ";B3
90  INPUT "BW     (HZ) = ";B1
100  IF B1 < B3 GOTO 140
110  PRINT : PRINT
120  PRINT "BW MUST BE LESS THAN ";B3;" HZ"
130  PRINT : PRINT : GOTO 90
140 T3 = 2 * K * B1 * T1 - B1 * B1
150 T2 = (T1 -  SQR (T1 * T1 - T3)) / 2
160 F1 = T2 + F2
170  PRINT "FM     = ";F1;" HZ"
180  PRINT : PRINT : PRINT
190  GOTO 90
```

Output example
Program then loops back to ask for another bandwidth.

```
]RUN
FM6/9
FX     (HZ) = 9677200
FINF   (HZ) = 9694000
K (TABLE 3) = 1.4142


MAX BW      = 6958.88223
BW     (HZ) = 3017
FM     = 9679517.59 HZ



BW     (HZ) =
```

Program 2. SSBBPF main program BPF frequency response calculator

```
 LOAD SSBBPF/4
]LIST

1  REM   SSB BPF FREQ RESPONSE(SSBBPF/4)
20  INPUT "FILTER ORDER = ";N
30  INPUT "FM           = ";F1
40  INPUT "BW           = ";B1
50  INPUT "FINF         = ";F9
60 B2 = B1 / 2:R1 = (F9 - F1) / B2: PRINT
100  INPUT "ATTENUATION(DB)   = ";D2
110  GOSUB 1000
120  PRINT "NORMALISED FREQ X = ";X1
130 J = ((R1 * X1) - 1) / (R1 - X1)
140  IF J < 0 THEN S = 0: GOTO 170
150 S = F1 - (B2 * J)
160  IF S < 0 THEN S = 0
170  PRINT "SLOW EDGE (HZ)    = ";S
180 L = ((R1 * X1) - 1) / (X1 - R1)
190  IF L < 0 THEN T4 = 0: GOTO 220
200 T4 = F1 + (B2 * L)
210  IF T4 < 0 THEN T4 = 0
```

```
220  PRINT "RETURN EDGE (HZ)  = ";T4
230 M = ((R1 * X1) + 1) / (R1 + X1)
240 U = F1 + (B2 * M)
250  PRINT "FAST EDGE (HZ)    = ";U
260  PRINT : PRINT : PRINT
270  PRINT "TO CONTINUE TYPE ATTENUATION VALUE"
280  PRINT "LESS THAN 200 DB. TO OBTAIN ULTIMATE"
290  PRINT "ATTENUATION TYPE 200": PRINT : PRINT
300  INPUT "ATTENUATION(DB)   = ";D2
310  IF D2 < 200 GOTO 110
320  GOSUB 1100
330  PRINT "ULTIMATE ATTENUATION (DB) = ";T3
340  PRINT "FREQ/BW RATIO R   = ";R1
350  PRINT : PRINT : END

]
```

Program 2(a). FRBUTT7 Butterworth subroutine for SSBBPF. Note the unusual type of "up"arrow used to denote exponentiation in line 1020 and subsequent lines and programs

```
]LIST

10  PRINT "FRBUTT7/3"
1000  REM  BUTTERWORTH FREQ RESPONSE
1010 D9 =  - D2 / 10:N9 = 1 / (2 * N)
1020 X1 = (1 / (10 ^ D9) - 1) ^ N9
1030  RETURN
1100  REM  ULTIMATE ATTENUATION
1110 V1 = 1 + R1 ^ (2 * N)
1120 T3 = 10 *  LOG (V1) /  LOG (10)
1130  RETURN

]
```

Program 2(b). FRCHEBY5 Chebyshev subroutine for SSBBPF

```
 LOAD FRCHEBY5/2
]LIST

10  PRINT "FRCHEBY5/2"
70  INPUT "PASSBAND RIPPLE(DB) = ";Y1
80  GOSUB 1200
1000  REM  CHEBYSHEV LPF FREQ RESPONSE
1010 A1 = 10 ^ (D2 / 20):A2 = 10 ^ (Y1 / 20)
1020 A3 =  SQR ((A1 ^ 2 - 1) / (A2 ^ 2 - 1))
1030 H2 = ( LOG (A3 +  SQR (A3 ^ 2 - 1))) / N
1040 X1 = ( EXP (H2) +  EXP ( - H2)) / (2 * P2)
1050  RETURN
1100  REM  ULTIMATE ATTENUATION
1110 V2 = A2 ^ 2 - 1:X2 = R1 * P2
1120 C1 =  LOG (X2 +  SQR (X2 ^ 2 - 1)) * N
1130 C2 = (( EXP (C1) +  EXP ( - C1)) / 2) ^ 2
1140 T2 = V2 * C2 + 1
1150 T3 = 10 *  LOG (T2) /  LOG (10)
1160  RETURN
1200  REM  CALC RIPPLE FACTOR
1210 E1 = 1 /  SQR (10 ^ (Y1 / 10) - 1)
1250  REM  CALC 2PI.F(3DB)
1260 P3 =  LOG (E1 +  SQR (E1 ^ 2 - 1)) / N
1270 P2 = ( EXP (P3) +  EXP ( - P3)) / 2
1280  RETURN

]
```

Output example from SSBBPF merged with FRBUTT7

```
]RUN
FRBUTT7/3
FILTER ORDER = 2
FM           = 9679518
BW           = 3017
FINF         = 9694000

ATTENUATION(DB)   = 40
NORMALISED FREQ X = 9.99975
SLOW EDGE (HZ)    = 0
RETURN EDGE (HZ)  = 10038249.7
FAST EDGE (HZ)    = 9686983.55


TO CONTINUE TYPE ATTENUATION VALUE
LESS THAN 200 DB. TO OBTAIN ULTIMATE
ATTENUATION TYPE 200


ATTENUATION(DB)   = 200
ULTIMATE ATTENUATION (DB) = 39.2918404
FREQ/BW RATIO R   = 9.60026517



]
```

Program 3. DISHAL4 filter component calculator

```
 LIST1-1200

10  PRINT "DISHAL4/4"
20  DIM C(4),K(4)
100  INPUT "INPUT FILTER ORDER 2,3,4,6 OR 8  = ";O1
110  IF O1 = 2 THEN S1 = 1:T1 = 2
115  IF O1 = 3 THEN S1 = 1:T1 = 2
120  IF O1 = 4 THEN S1 = 2:T1 = 2
125  IF O1 = 6 THEN S1 = 3:T1 = 3
130  IF O1 = 8 THEN S1 = 4:T1 = 4
140  INPUT "DISSIPATION COEFF D = ";D1
150  FOR S2 = 1 TO T1
160  PRINT "K";S2;S2 + 1;" = ";
170  INPUT "";K(S2): NEXT S2
200  INPUT "CS   (PF) = ";C2
210  INPUT "FM   (HZ) = ";F1
220  INPUT "FX   (HZ) = ";F2
230  INPUT "FINF (HZ) = ";F9
240  PRINT : PRINT
1000 M1 =  - (K(1) + K(2)) * (F9 - F2)
1010 N1 =  - (F2 - F1) * (F9 - F1)
1020 B1 =  - M1 -  SQR (M1 ^ 2 - (4 * N1))
1100 R1 = 2 * (F9 - F1) / B1
1110 R2 = 1 / R1
1120 R3 = R2 ^ 2
1130 P1 = 3.141593
1140 X3 = 1 / (2 * P1 * F2 * C2 * 1E - 12)
1150 X2 = X3 / ((1 - (R2 * (K(1) + K(2)))) ^ 2)
1160 R9 = X2 * D1 * B1 * (1 - R3) / F1
1170 C9 = C2 * F1 * (R1 - K(1) - K(2)) / ((R1 ^ 2 - 1) * B1)
1200  FOR S2 = 1 TO S1


]

 LIST1210-

1210 X1 = X2 * K(S2) * B1 * (1 - R3) / F1
1220 C(S2) = 1E12 / (2 * P1 * F1 * X1)
1230  PRINT "C";S2;"   (PF) = ";C(S2)
1240  NEXT S2
1300  IF O1 > 2 GOTO 1330
1310  PRINT "CS1  (PF) = ";C(1)
1320  GOTO 1500
1330  IF O1 > 3 GOTO 1360
1340  PRINT "CS1  (PF) = ";C(1)
1350  GOTO 1500
1360  IF O1 > 4 GOTO 1390
1370  PRINT "CS1  (PF) = ";C(2)
1380  GOTO 1500
1390  IF O1 > 6 GOTO 1440
1400  PRINT "CS1  (PF) = ";C(2)
1410 S3 = 1 / ((1 / C(1)) - (1 / C(3)))
1420  PRINT "CS3  (PF) = ";S3
1430  GOTO 1500
1440  PRINT "CS1  (PF) = ";C(2)
1450 S3 = 1 / ((1 / C(1)) - (1 / C(3)))
1460  PRINT "CS3  (PF) = ";S3
1470 S4 = 1 / ((1 / C(1)) + (1 / C(2)) - (1 / C(3)) - (1 / C(4)))
1480  PRINT "CS4  (PF) = ";S4
1500  PRINT "R         = ";R1
1510  PRINT "BW   (HZ) = ";B1
1520  PRINT "Z  (OHMS) = ";R9
1530  PRINT "CP   (PF) = ";C9
1540  END

]
```

Program 3 output example

```
]RUN
DISHAL4/4
INPUT FILTER ORDER 2,3,4,6 OR 8  = 2
DISSIPATION COEFF D = .7071
```

```
K12 = .7071
K23 = .7071
CS   (PF) = .03
FM   (HZ) = 9679518
FX   (HZ) = 9677200
FINF (HZ) = 9694000


C1   (PF) = 100.01006
CS1  (PF) = 100.01006
R         = 9.59871277
BW   (HZ) = 3017.48794
Z  (OHMS) = 164.407893
CP   (PF) = 8.64242679

]
```

Program 4. DANDK1 d and k calculator

```
]LIST

1  PRINT "DANDK1/3"
2  REM  CHEBYSHEV COUPLING
3  REM  COEFFICIENT CALCULATION
10  INPUT "RIPPLE  (DB) = ";Y1
20  INPUT "FILTER ORDER = ";O1
100  REM  CALC RIPPLE FACTOR
110 O2 = O1 - 1
120 R2 = Y1 / 10
130 E1 = 1 / ( SQR (10 ^ R2 - 1))
200  REM   EVALUATE SINH FUNCTION
210 S1 =  LOG (E1 +  SQR (E1 ^ 2 + 1))
220 S2 = S1 / O1
230 S3 = ( EXP (S2) -  EXP ( - S2)) / 2
300  REM   EVALUATE COSH FUNCTION
310 W1 =  LOG (E1 +  SQR (E1 ^ 2 - 1))
320 W2 = W1 / O1
330 W3 = ( EXP (W2) +  EXP ( - W2)) / 2
400  REM  CALC Q & D
410 Z = 3.14159 / 2
420 Q1 = 2 *  SIN (Z / O1) * W3 / S3
430 D1 = 1 / Q1
440  PRINT "DISSIPATION COEFF D = ";D1
500  PRINT : PRINT "COUPLING COEFFICIENTS": PRINT
510  FOR C1 = 1 TO O2
520 N1 = S3 ^ 2 + (( SIN (2 * C1 * Z / O1)) ^ 2)
530 D2 =  SIN ((2 * C1 - 1) * Z / O1)
540 D3 =  SIN ((2 * C1 + 1) * Z / O1)
550 D4 = 4 * D2 * D3
560 K1 = ( SQR (N1 / D4)) / W3
570  PRINT "K";C1;C1 + 1;"  = ";K1
580  NEXT C1
```

Output example

```
]RUN
DANDK1/3
RIPPLE  (DB) = 1
FILTER ORDER = 6
DISSIPATION COEFF D = .453494928

COUPLING COEFFICIENTS

K12  = .633495695
K23  = .531278928
K34  = .5201734
K45  = .531278789
K56  = .633494103
```

□

Notes on "An add-on capacitance measuring module for digital frequency counters"

by TONY BAILEY, G3WPO*

THE CAPACITY ADD-ON UNIT described in the September 1982 issue of *Radio Communication* appears to have been very popular judging by the correspondence which has resulted. Most of these letters have described problems that have resulted in getting the module going, and it was felt worthwhile to pass these on for the benefit of others who may have attempted construction but failed to finish the project for one or other of the following reasons.

7216 series ics

One of the major points has been an inability to achieve any results when using some samples of the 7216C/D integrated circuit. It appears that the required mip output on pin 2 was only added by the manufacturers in 1980, and prior to this date the pin was uncommitted! As the reference number for the ic was not changed, the only way of identifying this is by the date code which appears on the package. Anything starting with a year before 80 is definitely suspect, and, if 80, may be suspect. A large number of readily-obtainable units do employ this chip—if they are coded earlier than 1980, and have no measurable voltage at pin 2, the only course of action is to replace the chip (at around £20!).

The 7216A/B does not have any suitable outputs from which to take the gate pulse, and without considerable extra circuitry it is difficult to see how the unit can be used with these.

Other sources of the gate pulse

One point omitted from the original text was that if you are deriving the gate pulse from either the gate l.e.d., or a ttl voltage level source, then the resulting voltage obtained is likely to be insufficient to drive the circuit properly. A simple interface is required (Fig 1) which will give the correct voltage output. This circuit does invert the voltage, so the opposite input to the module will then be required.

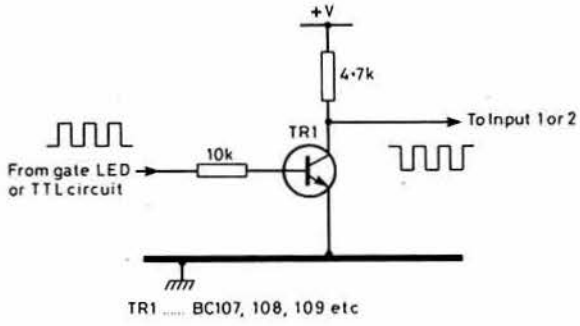

Fig 1. Interface circuit

DFM problems

A number of dfms have found difficulty in accepting the output of the unit and will not read correctly. The usual symptom is a brief correct reading followed by random or zero readings.

Part of the problem lies in the fact that the unit's output is a square wave, and most dfms prefer or are designed for a sine wave input. Also the dfm can be suffering from pure overload. In practice, the first problem can be overcome by adding a capacitor of around 330pF across the unit's output, and the second by the addition of a series resistor in the output, value from 100–470kΩ, sometimes both being needed.

Miscellaneous

It is important that the ics specified in the parts list are used—using buffered instead of unbuffered cmos, and vice versa, will result in failure (suffix A or UB types are unbuffered, B are buffered).

When constructing the unit, keep the leads from the terminals which take the capacitor under test to the pcb as short as possible, as otherwise the residual capacity may be in excess of that which the zeroing circuit can accommodate. □

*20 Farnham Avenue, Hassocks, W. Sussex BN6 8NS.